

ROBERT A. CESARI (1928-2008)
JOHN F. MCKENNA
MARTIN J. O'DONNELL
THOMAS C. O'KONSKI
PATRICIA A. SHEEHAN
MICHAEL E. ATTAYA
CHARLES J. BARBAS
MICHAEL R. REINEMANN
KEVIN GANNON
DUANE H. DREGER
JAMES A. BLANCHETTE
JAMES M. BEFMKE
SHANNEN C. DELANEY
OMAR M. WADHWA
RITA M. ROONEY
MICHAEL T. ABRAMSON
STEPHEN D. LEBARRON

CESARI AND MCKENNA, LLP
ATTORNEYS AT LAW
88 BLACK FALCON AVENUE
BOSTON, MASSACHUSETTS

Telephone: (617) 951-2500 Telecopier: (617) 951-3927
Website: www.c-m.com

INTELLECTUAL PROPERTY
AND RELATED
CAUSES

A. SIDNEY JOHNSTON
EDWIN H. PAUL
OF COUNSEL

HEATHER SHAPIRO
PATENT AGENT

FACSIMILE COVER SHEET

112056-0474

DATE:	April 1, 2009
TOTAL PAGES WITH COVER:	20
TO:	Giovanna B. Colan
FIRM:	United States Patent and Trademark Office
FACSIMILE NUMBER:	571-273-2752
TELEPHONE NUMBER:	571-272-2752
FROM:	Michael T. Abramson

COMMENTS:

Please call Kristin at 617-951-3089 to confirm receipt of this agenda.

Thank you!

SPECIAL INSTRUCTIONS:

If you do not receive all pages, or you are not the intended recipient, please contact us at (617) 951-2500 as soon as possible.

PATENTS
112056-0474
P01-2475.01

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re The Application of:
Christo Iankov Bojinov

Serial No.: 10/803,788

Filed: March 17, 2004

For: METHOD AND APPARATUS
FOR IMPROVING FILE SYSTEM
PROXY PERFORMANCE AND
SECURITY BY DISTRIBUTING
INFORMATION TO CLIENTS VIA
FILE HANDLES

Examiner: Colan, Giovanna B

Art Unit: 2162

Confirmation No.: 8050

AGENDA FOR INTERVIEW

The Agenda is:

- (1) Explain the problem solved
- (2) Analyze the claimed solution
- (3) Analyze all cited art
- (4) Explain why all claims are allowable in view of the cited prior art

Present Status of Case

This Agenda is sent in response to the Office Action mailed by USPTO on
February 23, 2009.

Claims: All claims 1-5 and 30-65 have been rejected.

- This Agenda is sent via facsimile to Examiner Colan.
 - Facsimile #: (571)-273-2752
- Agenda for a telephonic interview with Examiner Giovanna B. Colan.
- Attorney Michael T. Abramson (Reg. No. 60,320) will call Examiner Colan for the scheduled interview on Monday April 6, 2009 at 10:00 A.M. (EST).
 - Telephone #: (571)-272-2752

PATENTS
112056-0474
P01-2475.01

PROPOSED CLAIMS:

- 1 1. (Proposed Amended) A method for establishing identity in a file system, comprising:
2 receiving a file request concerning an indicated file from a client, the request
3 received by a proxy;
4 forwarding the request from the proxy to a file server;
5 returning a reply associated with the file request from the file server to the proxy;
6 inserting, by the proxy, metadata into the file handle; and
7 sending, by the proxy, the file handle with the metadata inserted in the file handle
8 to the client, the metadata to be used in further requests to identify the client and the
9 indicated file.
- 1 2. (Previously Presented) The method of Claim 1, further comprising:
2 using the metadata in the file handles for any of eliminating a need for the proxy
3 to generate additional requests to the server to establish file identity, and for completing
4 client requests.
- 1 3. (Previously Presented) The method of Claim 1, further comprising:
2 encoding metadata in a form of a session key into the file handle, the session key
3 expiring after a predetermined amount of time.
- 1 4. (Previously Presented) The method of Claim 1, further comprising:
2 using an NFS file system as the file system.
- 1 5. (Previously Presented) The method of Claim 1, further comprising:
2 using a stateless protocol by the file system.
- 1 6-29. (Cancelled).
- 1 30. (Previously Presented) The method of claim 1, further comprising:

PATENTS
112056-0474
P01-2475.01

receiving, from the client, a second file request by the proxy, the second file request including the metadata in a further file handle sent with the second request; identifying, in response to the metadata, the client as having a permission to submit the second file request; sending the second file request to the file server and not sending the metadata with the second file handle to the file server; and receiving by the proxy the further reply from the file server, and sending by the proxy the further reply to the client.

31. (Proposed Amended) A method for establishing identity in a file system, comprising:

receiving a first file request concerning an indicated file from a client, the first file request received by a proxy; forwarding the first file request from the proxy to a file server; returning a reply associated with the first file request from the file server to the proxy, wherein the reply includes a file handle associated with the indicated file; inserting, by the proxy, metadata into the file handle; sending, by the proxy, the file handle with the metadata inserted in the file handle to the client, the metadata to be used in further requests to identify the client as having a permission to access the indicated file; receiving, from the client, a second file request by the proxy, the second file request including the metadata in a second file handle sent with the second file request; identifying, in response to the metadata, that the client has the permission to submit the second file request; sending the second file request to the file server and not sending the metadata with the second file handle to the file server; and receiving by the proxy a second reply from the file server, and sending by the proxy the second reply to the client.

PATENTS
112056-0474
P01-2475.01

32. (Proposed Amended) An apparatus to establish identity in a file system,
comprising:

a proxy to receive a file request sent by a client to ~~a~~ the file system, the proxy to
forward the request to a file server;

the file server to return a reply associated with the file request to the proxy,
wherein the reply includes a file handle;

the proxy to insert metadata into the file handle; and

the proxy to send the file handle with the metadata inserted in the file handle to
the client, the metadata to be used in further requests to identify the client and the
indicated file.

33. (Previously Presented) The apparatus as in claim 32, further comprising:

the proxy to receive, by the client, a second file request, the second file request to
include the metadata in the second file handle sent with the second request;

the proxy to identify, in response to the metadata, the client as having a
permission to submit the second file request;

the proxy to send the second file request to the file server and not to send the
metadata with the second file handle to the file server; and

the proxy to receive a second reply from the file server, and the proxy to send the
second reply to the client.

34. (Previously Presented) The apparatus of Claim 32, further comprising:

the proxy to use the metadata in the file handle received from the client to
eliminate a need for additional communication with the file server to establish file
identity.

35. (Previously Presented) The apparatus of Claim 32, further comprising:

the proxy to encode the metadata in a form of a session key into the file handle,
the session key expiring after a predetermined amount of time.

PATENTS
112056-0474
P01-2475.01

1 36. (Previously Presented) The apparatus of Claim 32, further comprising:
2 an NFS file system used as the file system.

1 37. (Previously Presented) The apparatus of Claim 32, further comprising:
2 a stateless protocol used by the file system.

1 38. (Proposed Amended) A non-volatile memory executed on a computer, comprising:
2 said non-volatile memory containing procedures for execution on the computer
3 for a method of establishing identity in a file system, the method having the steps of,
4 receiving a file request concerning an indicated file from a client, the request
5 received by a proxy;
6 forwarding the request from the proxy to a file server;
7 returning a reply associated with the file request from the file server to the proxy,
8 wherein the reply includes a file handle associated with the indicated file;
9 inserting, by the proxy, metadata into the file handle; and
10 sending, by the proxy, the file handle with the metadata inserted in the file handle
11 to the client, the metadata to be used in further requests to identify the client and the
12 indicated file.

1 39. (Proposed Amended) A method for establishing identity in a file system,
2 comprising:
3 receiving a first file request concerning an indicated file from a client, the first file
4 request received by a proxy;
5 forwarding the first file request from the proxy to a file server;
6 granting a permission for the request to be acted upon by the file system in
7 response to a predetermined protocol;
8 returning a reply associated with the first file request from the file server to the
9 proxy, wherein the reply includes a file handle associated with the indicated file;
10 inserting, by the proxy, a session key into the file handle; and

PATENTS
112056-0474
P01-2475.01

11 sending, by the proxy, the file handle with the session key inserted in the file
12 handle to the client, the session key to be used in further requests to identify the client
13 and the indicated file.

1 40. (Previously Presented) The method according to claim 39, further comprising:
2 receiving, from the client, a second file request by the proxy, the second file
3 request including the session key in a second file handle sent with the second file request;
4 identifying, in response to the session key, that the client has the permission to
5 submit the second file request;
6 sending the second file request to the file server and not sending the session key
7 with the second file handle to the file server; and
8 receiving by the proxy a second reply from the file server, and sending by the
9 proxy the second reply to the client.

1 41. (Previously Presented) The method according to claim 39, further comprising:
2 causing the session key to expire after a selected amount of time.

1 42. (Previously Presented) The method according to claim 39, further comprising:
2 causing the session key to expire after a selected amount of usage.

1 43. (Previously Presented) The method according to claim 39, further comprising:
2 using a NFS protocol as the predetermined protocol.

1 44. (Previously Presented) The method according to claim 43, further comprising:
2 using as the predetermined protocol a two way communication exchange between
3 the proxy and the file server.

1 45. (Proposed Amended) An apparatus to establish identity in a file system, comprising:
2 a proxy to receive a file request sent by a client to a the file system, the proxy to
3 forward the request to a file server;

PATENTS
112056-0474
P01-2475.01

the file server to return a reply associated with the file request to the proxy,
wherein the reply includes a file handle;
the proxy to insert a session key into the file handle; and
the proxy to send the file handle with the session key inserted in the file handle to
the client, the session key to be used in further requests to identify the client and the
indicated file.

46. (Previously Presented) The apparatus as in claim 45, further comprising:
the proxy to receive, by the client, a second file request, the second file request to
include the session key in a further file handle sent with the second request;
the proxy to identify, in response to the session key, the client as having a
permission to submit the another file request;
the proxy to send the second request to the file server and not to send the session
key with the second file handle to the file server; and
the proxy to receive a further reply from the file server, and the proxy to send the
further reply to the client.

47. (Previously Presented) The apparatus of Claim 45, further comprising:
the proxy to use the metadata in the file handle received from the client to
eliminate a need for additional communication with the file server to establish file
identity.

48. (Previously Presented) The apparatus of Claim 45, further comprising:
the proxy to encode the metadata in a form of a session key into the file handle,
the session key expiring after a predetermined amount of time.

49. (Previously Presented) The apparatus of Claim 45, further comprising:
an NFS file system used as the file system.

50. (Previously Presented) The apparatus of Claim 45, further comprising:

PATENTS
112056-0474
P01-2475.01

2 a stateless protocol used by the file system.

1 51. (Proposed Amended) An apparatus to establish identity in a file system, comprising:
2 a proxy configured to receive a first file request sent by a client to ~~a~~ the file
3 system, the proxy further configured to forward the first file request to a file server;
4 the file server configured to return a reply associated with the first file request to
5 the proxy;
6 the proxy further configured to insert a session key into a file handle;
7 the proxy further configured to send the file handle with the session key inserted
8 in the file handle to the client, the session key configured to be used in a second file
9 request to identify the client and the indicated file;
10 the proxy further configured to receive, by the client, a second file request, the
11 second file request configured to include the session key in a second file handle sent with
12 the second file request;
13 the proxy further configured to identify, in response to the session key, the client
14 as having a permission to submit the second file request;
15 the proxy further configured to send the second file request to the file server and
16 not to send the session key with the second file handle to the file server; and
17 the proxy further configured to receive a second reply from the file server, and the
18 proxy further configured to send the second reply to the client.

1 52. (Proposed Amended) A method for establishing identity in a file system,
2 comprising:
3 receiving a first file request concerning an indicated file from a client, the first file
4 request received by a proxy;
5 forwarding the first file request from the proxy to a file server;
6 determining that the client has a permission to have the request acted upon by the
7 file system in response to a predetermined protocol;
8 returning a reply associated with the first file request from the file server to the
9 proxy, wherein the reply includes a file handle associated with the indicated file;

PATENTS
112056-0474
P01-2475.01

10 inserting, by the proxy, a cryptographic information into the file handle;
11 sending, by the proxy, the file handle with the cryptographic information inserted
12 in the file handle to the client, the cryptographic information to be used in one or more
13 requests to identify the client and the indicated file.

1 53. (Previously Presented) The method according to claim 52, further comprising:
2 receiving, by the client, a second file request by the proxy, the second file request
3 including the cryptographic information in a second file handle sent with the second file
4 request;
5 identifying, in response to the cryptographic information, that the client has the
6 permission to submit the second file request;
7 sending the second file request to the file server and not sending the cryptographic
8 information with the second file handle to the file server; and
9 receiving by the proxy a second reply from the file server, and sending by the
10 proxy the second reply to the client.

1 54. (Previously Presented) The method according to claim 52, further comprising:
2 causing the cryptographic information to expire after a selected amount of time.

1 55. (Previously Presented) The method according to claim 52, further comprising:
2 causing the cryptographic information to expire after a selected amount of usage.

1 56. (Previously Presented) The method according to claim 52, further comprising:
2 using a NFS protocol as the predetermined protocol.

1 57. (Previously Presented) The method according to claim 52, further comprising:
2 using as the predetermined protocol a two way communication exchange between
3 the proxy and the file server.

1 58. (Proposed Amended) An apparatus to establish identity in a file system, comprising:

PATENTS
112056-0474
P01-2475.01

a proxy configured to receive a file request for an indicated file sent by a client to
a the file system, the proxy further configured to forward the request to a file server;
the file server configured to return a reply associated with the file request to the
proxy, wherein the reply is configured to include a file handle;
the proxy further configured to insert a cryptographic information into the file
handle; and

the proxy further configured to send the file handle with the cryptographic
information inserted in the file handle to the client, the cryptographic information
configured to be used in further requests to identify the client and the indicated file.

59. (Proposed Amended) The apparatus as in claim 58, further comprising:

the proxy further configured to receive, by the client, a second request, the second
file request to include the cryptographic information in a second file handle sent with the
second request;

the proxy further configured to identify, in response to the cryptographic
information, the client as having a permission to submit the second file request;

the proxy further configured to send the second request to the file server and not
to send the cryptographic information with the second file handle to the file server; and

the proxy further configured to receive a further reply from the file server, and the
proxy to send the further reply to the client.

60. (Proposed Amended) The apparatus of claim 58, further comprising:

the proxy further configured to use the metadata in the file handle received from
the client to eliminate a need for additional communication with the file server to
establish file identity.

61. (Proposed Amended) The apparatus of claim 58, further comprising:

the proxy further configured to encode the metadata in a form of a cryptographic
information into the file handle, the cryptographic information configured to
expire ~~expiring~~ after a predetermined amount of time.

PATENTS
112056-0474
P01-2475.01

1 62. (Previously Presented) The apparatus of claim 58, further comprising:
2 an NFS file system used as the file system.

1 63. (Previously Presented) The apparatus of claim 58, further comprising:
2 a stateless protocol used by the file system.

1 64. (Proposed Amended) An apparatus to establish identity in a file system,
2 comprising:
3 a proxy configured to receive a first file request sent by a client to ~~a~~the file
4 system, the proxy to forward the first file request to a file server;
5 the file server configured to return a reply associated with the first file request
6 to the proxy;
7 the proxy further configured to insert a cryptographic information into a file
8 handle;
9 the proxy further configured to send the file handle with the cryptographic
10 information inserted in the file handle to the client, the cryptographic information
11 configured to be used in a second file request to identify the client and the indicated
12 file;
13 the proxy further configured to receive, by the client, a second file request, the
14 second file request configured to include the cryptographic information in a second
15 file handle sent with the second file request;
16 the proxy further configured to identify, in response to the cryptographic
17 information, the client as having a permission to submit the second file request;
18 the proxy further configured to send the second file request to the file server
19 and not to send the cryptographic information with the second file handle to the file
20 server; and
21 the proxy further configured to receive a second reply from the file server, and
22 the proxy to send the second reply to the client.

PATENTS
112056-0474
P01-2475.01

1 65. (Proposed Amended) A method for establishing identity in a file system,
2 comprising:
3 receiving a file request concerning an indicated file from a client, the request
4 received by a proxy;
5 forwarding the request from the proxy to a file server;
6 returning a reply associated with the file request from the file server to the
7 proxy, wherein the reply includes a file handle associated with the indicated file;
8 inserting, by the proxy, metadata into the file handle; and
9 sending, by the proxy, the file handle with the metadata inserted in the file
10 handle to the client, a size of the file handle set to a sum of a length of the server file
11 handle and a length of the proxy metadata, the metadata to be used in further requests
12 to identify the client and the indicated file.

PATENTS
112056-0474
P01-2475.01

REMARKS

Claims 1-5 and 30-65 are currently pending in the case.

A new claim 66 is being proposed.

A proposed amendment to the claims, in particular claim 1, is included.

PROBLEM SOLVED

The Network File System (NFS) is a stateless protocol. This means that the file server stores no per-client information, and there are no NFS connections. For example, NFS has no operation to open a file because this would require the server to store state information, e.g. that a file is open, what its file descriptor is, the next byte to read, etc. Instead, NFS supports a lookup procedure, which converts a filename into a file handle. This file handle is a unique, immutable identifier, usually an inode number, or disk block address. NFS does have a read procedure, but the client must specify a file handle and starting offset for every call to read.

A software program or appliance that is a proxy for the NFS protocol, or any other protocol that uses server-generated file handles, usually requires additional file metadata information to be stored either on the server or locally on the proxy. This metadata can be used, for example, to apply different encryption keys, or to enforce access restrictions to files that are located in different logical units that are defined on the proxy, but possibly invisible to the file server.

Such an appliance forwards file handles generated by the file server to clients, and subsequently acts as a proxy for client requests for access to the file system on the server. Given a file handle from a client, the appliance needs to establish to what area (a.k.a. storage vault) the file belongs, and use the appropriate keys to encrypt or decrypt data. If the metadata used to establish this are not available on the proxy, as is typically the case with large file sets accessed by many client machines, the proxy must send additional requests to the file server to determine how to handle the client request correctly.

It would be advantageous to provide a mechanism that distributes information, effectively caches information, and provides a mechanism that improves performance by eliminating the need for the proxy to generate additional requests to the server to

PATENTS
112056-0474
P01-2475.01

establish file identity.

Rejections Under 35 U.S.C. § 103

At Paragraph 7 of the Office Action, claims 1-2, 4-5, 30-34, 36-40, 42-47, 49-53, 56-60, 62-64, and 65 were rejected under 35 U.S.C. 103(a) as being anticipated by Chandrashekar et al., U. S. Patent Publication 2005/0033988 published on February 10, 2005 (hereinafter "Chandrashekar"), and in view of Gvily, U.S. Patent Application Publication No. 2002/0078201 published on June 20, 2002 (hereinafter "Gvily").

Applicant's claimed novel invention, as set out in representative claim 1, comprises in part:

1. A method for establishing identity in a file system, comprising:
receiving a file request concerning an indicated file from a client,
the request received by a proxy;
forwarding the request from the proxy to a file server;
returning a reply associated with the file request from the file server to the proxy;
inserting, by the proxy, metadata into a file handle; and
sending, by the proxy, the file handle with the metadata inserted in the file handle to the client, the metadata to be used in further requests to identify the client and the indicated file.

Chandrashekar discusses processing file requests sent by a client and received by a proxy using security applications to encrypt, decompress, verify, and decrypt network data by a server receiving the files from the proxy [0058; 0071]. Header policy information is determined, generated, and added at the proxy server and then stored on the filer server [0055; Fig. 4-5]. Metadata of the header policy information now being sent by the server back to the proxy is stripped from the file (by the proxy) before the file is returned to the client [0038; Fig. 8; 0055; 0069-0070].

Gvily discusses embedding data in a text page at a proxy, but more particularly, embedding either meta-data or scripts into Hypertext Markup Language (HTML) (text) pages. The intermediary proxy analyzes the unstructured data of the HTML (text) page,

PATENTS
112056-0474
P01-2475.01

understands the meaning behind the data, associates meta-data with some of the unstructured data, and stores this meta-data back into the original HTML (text) page [0018; 0005]. The meta-data of understood data will be embedded into the HTML (text) page, effectively altering the source code [0035-0036; see also Figs. 4-5].

Applicant respectfully urges that Chandrashekhhar, taken singly or in any combination with Gvily, does not disclose Applicant's claimed novel use of **inserting metadata into a file handle** and then having the proxy send *the file handle with the metadata inserted in the file handle to the client*.

Applicant claims **inserting metadata into a file handle**. For example, when the client issues a file request to the server, the server may issue a file handle to the client. The file handle may be intercepted by a proxy, where instead of returning the file handle(s) to the client as-is, additional information (e.g., server name/ID, time, temporary session key, other cryptographic information, etc.) may be encoded into the handle. Applicant then claims **sending, by the proxy, the file handle with the metadata inserted in the file handle to the client**.

As stated above and stated by Examiner in the Office Action, Chandrashekhhar *removes the metadata* from the file data/file attributes (e.g., file handle) *before* returning the file to the client, and thus does not show Applicant's claimed **sending, by the proxy, the file handle with the metadata inserted in the file handle to the client**. However, more importantly, Chandrashekhhar explicitly *teaches away* from Applicant's claimed **sending, by the proxy, the file handle with the metadata inserted in the file handle to the client** since Chandrashekhhar *removes the metadata* from the file data/file attributes *before* returning the file to the client as a safety measure. As such, because Chandrashekhhar removes the metadata before returning the file to the client, there would be no motivation to combine Chandrashekhhar with any potential reference. As a result, in addition to Chandrashekhhar being totally silent to Applicant's claimed **sending, by the proxy, the file handle with the metadata inserted in the file handle to the client**,

PATENTS
112056-0474
P01-2475.01

Chandrashekhkar is also legally precluded from being used as a reference under 35 U.S.C. §103.

As noted above, Gvily shows embedding meta-data into an Hypertext Markup Language (HTML (text) page). Applicant respectfully argues that embedding meta-data into an HTML (text) page is not the same as Applicant's claimed **inserting metadata into a file handle**. Specifically, in one example, embedding meta-data into an HTML (text) page would require altering the HTML (text) page's source code, as is stated by Gvily at paragraph 36. This is because, as is well known in the art, HTML text is used to enable the web pages themselves to be more animated and interactive. In contrast, Applicant's **inserting metadata into a file handle** does not require altering the requested file. Put another way, an HTML (text) page is not the same as a file handle. Therefore, Gvily does not show Applicant's claimed **inserting metadata into a file handle** because an HTML (text) page is not the same as a *file handle*.

Accordingly, Applicant respectfully urges that Chandrashekhkar, taken singly or in any combination with Gvily, is legally insufficient to render the presently claimed invention obvious under 35 U.S.C. § 103. Chandrashekhkar and Gvily, taken singly or in any combination, does not disclose Applicant's claimed novel **inserting metadata into a file handle** and then having the proxy send *the file handle with the metadata inserted in the file handle to the client*.

Applicant's Interpretation of the Prior Art

Applicant's interpretation of the prior art references was derived, in part, from the following excerpts:

Chandrashekhkar

[0038]...The meta-data relates to key management, length of the original file/dataset, whether the file was compressed prior to encryption or not, integrity checks for file data. The meta-data is stripped off before the file data/file attributes are returned to the client... (emphasis added)

Gvily

PATENTS
112056-0474
P01-2475.01

[0018] The invention described hereinafter discloses a computer implemented method for embedding data in a text page at a proxy, where the proxy is generally an intermediary between a resource and a request for a resource. More particularly, the invention provides exemplary systems and methods for embedding either meta-data or scripts into HTML pages by means of a proxy. The intermediary proxy analyzes the unstructured data of a hypertext page, understands the meaning behind the data, associates meta-data with some of the unstructured data and stores this meta-data back into the original hypertext page. The invention potentially stores meta-data in a location that is hidden from the user's view so that it is unobtrusive but easily retrievable. (emphasis added)

[0034] Embedding Data Examples

[0035] FIG. 4 shows an example of a prior art hypertext page 401 consisting of some text and a hyperlink 402. The following illustrates source code that may be used to render that page:

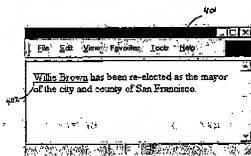


Figure 4 - Prior Art

```
1 <HTML> <HEAD> </HEAD> <BODY> <A
  HREF="http://xyz.somewhe-re.com">Willie Brown</A> has been re-
  elected as the mayor of the city and county of San Francisco. </BODY>
</HTML>
```

[0036] For purposes of the example, assume that the analysis process for the HTML page has recognized two objects on the page: Willie Brown and San Francisco. Willie Brown is recognized as a name, Willie as a first name and Brown as a last name. Continuing, San Francisco is recognized as a location. The meta-data of these objects will be embedded into the web page, effectively altering the source code to something like:

PATENTS
112056-0474
P01-2475.01

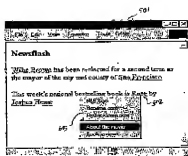


Figure 1

```
2 <HTML> <HEAD> </HEAD> <BODY> <A
  HREF="http://xyz.somewhere.com"
  META="<PERSON><FIRST>Willie</FIRST>
    <LAST>Brown</LAST></PERSON>">Willie Brown</A> has been re-
    elected as the mayor of the city and county of <SPAN
      META="<LOCATION><CITY>San
        Francisco</CITY><STATE>CA</STATE></LOCATION>">- San
        Francisco</SPAN>. </BODY> </HTML>
```

(emphasis added to [0035 and 0036])

Rejections Under 35 U.S.C. § 103

At Paragraph 8 of the Office Action, claims 3, 35, 41, 48, 54-55, and 61 were rejected under 35 U.S.C. 103(a) as being unpatentable over Chandrasekhar in view of Gvily, and in further view of Ohazama et al., U. S. Patent No. 7,225,207 issued on May 29, 2007 (hereinafter "Ohazama").

Applicant respectfully notes that claims 3, 35, 41, 48, 54-55, and 61 are all dependent claims, and these dependent claims are dependent from independent claims which are believed to be in condition for allowance. Accordingly, claims 3, 35, 41, 48, 54-55, and 61 are believed to be in condition for allowance.

Proposed New Claim

Any currently proposed claim(s), whether amended or new, are believed to be in condition for allowance and fully supported by Applicant's specification

PATENTS
112056-0474
P01-2475.01

Applicant's claimed novel invention, as set forth in representative new claim 66, comprises in part:

66. A method, comprising:
receiving, by a proxy, a file request for a file sent from a client;
forwarding the file request from the proxy to a file server;
returning a reply associated with the file request from the file server to the proxy, wherein the reply includes a file handle;
inserting, by the proxy, metadata into the file handle;
sending, by the proxy, the file handle with the metadata inserted in the file handle to the client; and
using, by the client, the metadata inserted into the file handle in a subsequent file request to identify the client and the file.

Conclusion

All newly proposed claims and proposed claim amendments are believed to be fully supported by Applicant's specification.

All proposed independent claims are believed to be in condition for allowance.

All proposed dependent claims are believed to be dependent from allowable proposed independent claims, and therefore in condition for allowance.